International Journal of Applied Sciences and Innovation
Vol. 2015, no. 1, pp. 10-16
ISSN: 2458-6501

Derivative Free Cubic Convergent Extrapolated Newton's Method

V.B. Kumar Vatti[1], Shouri Dominic[1] and Sahanica,V[2]

[1]Department of Engineering Mathematics, Visakhapatnam, Andhra Pradesh, India
[2]ROLTA India Ltd[3], Mumbai, India
Email: *drvattivbk@yahoo.co.in*

Abstract- In this paper we consider the two step process cubic convergent extrapolated Newton's method which requires the evaluation of the first derivative and suggest a two-step process of the same method free from the derivatives by using backward difference approximation. It is shown that the new method has a cubic rate of convergence as that of Extrapolated Newton's method and the efficiency index of this method is $\sqrt[3]{3}$. Few examples are given to illustrate the efficiency of this method compared to Newton's method and extrapolated Newton's method.

Keywords: *Nonlinear equations ; Iterative method ; Convergence Criteria ; Newton's Method ; Cubic Convergence ;.*

I. INTRODUCTION

We consider finding the zeros of a nonlinear equation

$$f(x) = 0 \tag{1.1}$$

Where $f : D \subset R \to R$ is a scalar function on an open interval D and $f(x)$ may be algebraic, transcendental or combined of both.

The famous well known quadratic convergent Newton's method for finding the root $'\alpha'$ of (1.1) is given by

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \quad (n = 0, 1, 2,\dots) \tag{1.2}$$

Starting from the initial approximation $'x_0'$ which is in the vicinity of the exact root, V.B. Kumar, Vatti et. al. [15] suggested a cubic convergent method by extrapolating the newton's method (1.2) known as extrapolated Newton's method which requires three functional evaluations i.e; $f(x)$, $f'(x)$ and $f''(x)$ in each iteration, defined by

$$x_{n+1} = x_n - \bar{\alpha}.\frac{f(x_n)}{f'(x_n)} \quad (n= 0, 1, 2...) \tag{1.3}$$

Here $'\bar{\alpha}'$ is the relaxation parameter and its optimal choice given by

$$\bar{\alpha} = \frac{2}{2 - \rho_n} \tag{1.4}$$

where $$\rho_n = \frac{f(x_n) f''(x_n)}{f'^2(x_n)} \qquad (1.5)$$

A variant of Newton's method suggested by Fernando and Weerakon [13] which is a two step iterative process, defined by

$$x_{n+1} = x_n - \frac{2f(x_n)}{f'(x_n) + f'(x^*_{n+1})} \qquad (n= 0, 1, 2...) \qquad (1.6)$$

where $$x^*_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

This method (1.6) has a third order convergence.

It is to note that the efficiency indexes of the methods (1.2), (1.3), and (1.6) are $\sqrt{2}, \sqrt[3]{3}$ and $\sqrt[3]{3}$ respectively. In this paper, we present the method (1.3) as a two-step iterative process free from all the derivatives in section 2. In section 3, the convergence criteria of the new method is discussed where as in the concluding section several numerical examples are considered to exhibit the efficiency of the developed method.

II. DERIVATIVE FREE EXTRAPOLATED NEWTON'S METHOD

The cubic convergent extrapolated Newton's method free from second derivative suggested by V.B. Kumar,Vatti et. al. [16] is given as:

For a given x_0, compute the approximate solution x_{n+1} by iterative scheme:

$$x_{n+1} = x_n - \left[\frac{f(x_n)}{f(x_n) - f(y_n)}\right] \frac{f(x_n)}{f'(x_n)} \qquad (n=0, 1, 2...) \qquad (2.1)$$

Where $$y_n = x_n - \frac{f(x_n)}{f'(x_n)} \qquad (2.2)$$

As it is known that the backward difference approximation for the first derivative for $f'(x)$ at *x* is

$$f'(x) \approx \frac{f(x) - f(x-h)}{h} \qquad (2.3)$$

Replacing '*h*' by $f(x_n)$, the backward difference approximation for the derivative $f'(x_n)$ at x_n is

$$f'(x_n) \approx \frac{f(x_n) - f(x_n - f(x_n))}{f(x_n)} \qquad (2.4)$$

Now the method (2.1) takes the form

$$x_{n+1} = x_n - \frac{f^2(x_n)}{f(x_n) - f(x_n - f(x_n))} \left[\frac{f(x_n)}{f(x_n) - f(y_n)}\right] \text{(n=0, 1, 2...)} \qquad (2.5)$$

where $$y_n = x_n - \frac{f^2(x_n)}{f(x_n) - f(x_n - f(x_n))}$$

We now define the following algorithm.

Algorithm 2.1: For a given x_0, compute the approximate solution x_{n+1} by iterative scheme:

$$x_{n+1} = x_n - \frac{f^2(x_n)}{f(x_n)-f(x_n-f(x_n))}\left[\frac{f(x_n)}{f(x_n)-f(y_n)}\right] \text{(n=0, 1, 2...)}$$

where $$y_n = x_n - \frac{f^2(x_n)}{f(x_n)-f(x_n-f(x_n))}$$

This algorithm can be called as a two step derivative free extrapolated Newton's method and requires three functional evaluations. The efficiency index of this method is $\sqrt[3]{3}$.

III. CONVERGENCE CRITERIA

Theorem 3.1: *Let $\alpha \in D$ be a simple zero of the function $f : D \subset R \to R$ for an open interval D. If x_0 is in the vicinity of exact root α , then the algorithm (2.1) has third order convergence and it satisfies the error equation.* $e_{n+1} = \left[\frac{c_2^2}{c_1}\left(\frac{1}{c_1}-1\right)\right]e_n^3$ *where* $e_n = x_n - \alpha$.

Proof: If α be the root of $f(x)=0$ and

$$e_n = x_n - \alpha \tag{3.1}$$

be the error at n[th] iteration. Then by Taylor's series, we have

$$f(x_n) = f(\alpha + e_n)$$

$$= f(\alpha) + f'(\alpha)e_n + e_n^2\frac{f''(\alpha)}{2!} + e_n^3\frac{f'''(\alpha)}{3!} + e_n^4\frac{f^{iv}(\alpha)}{4!} + e_n^5\frac{f^{V}(\alpha)}{5!} + O(e_n^6)$$

$$= c_1e_n + c_2e_n^2 + c_3e_n^3 + c_4e_n^4 + c_5e_n^5 + O(e_n^6) \tag{3.2}$$

Where

$$c_j = \frac{f^j(\alpha)}{j!} \tag{3.3}$$

and,

$$= (-c_1^2 + c_1)e_n + (c_2c_1^2 - 3c_2c_1 + c_2)e_n^2 + (-c_3c_1^3 + 3c_3c_1^2 + 2c_1c_2^2 - 4c_3c_1 - 2c_2^2 + c_3)e_n^3$$
$$+ (c_4 + c_2(c_2^2 + 2c_1c_3) - 5c_1c_4 - 5c_2c_3 + 6c_1^2c_4 - 4c_1^3c_4 + c_1^4c_4 + 6c_1c_2c_3 - 3c_1^2c_2c_3)e_n^4 + O(e_n^5) \tag{3.4}$$

Subtracting (3.2) from (3.4) we get

$$f(x_n) - f(x_n - f(x_n))$$
$$= c_1^2e_n + (-c_2c_1^2 + 3c_2c_1)e_n^2 + (c_3c_1^3 - 3c_3c_1^2 - 2c_1c_2^2 + 4c_3c_1 + 2c_2^2)e_n^3$$
$$+ (-c_2(c_2^2 + 2c_1c_3) + 5c_1c_4 + 5c_2c_3 - 6c_1^2c_4 + 4c_1^3c_4 - c_1^4c_4 - 6c_1c_2c_3 + 3c_1^2c_2c_3)e_n^4 + O(e_n^5) \tag{3.5}$$

Also $$[f(x_n)]^2 = [c_1e_n + c_2e_n^2 + c_3e_n^3 + c_4e_n^4 + c_5e_n^5 + O(e_n^6)]^2$$
$$= c_1^2e_n^2 + 2c_1c_2e_n^3 + (c_2^2 + 2c_1c_3)e_n^4 + (2c_2c_3 + 2c_1c_4)e_n^5 \tag{3.6}$$

From (3.5) and (3.6) we get

$$\frac{[f(x_n)]^2}{f(x_n)-f(x_n-f(x_n))}$$

$$=\frac{c_1^2e_n^2+2c_1c_2e_n^3+(c_2^2+2c_1c_3)e_n^4+(2c_2c_3+2c_1c_4)e_n^5+O(e_n^6)}{\begin{array}{l}\{c_1^2e_n+(-c_2c_1^2+3c_2c_1)e_n^2+(c_3c_1^3-3c_3c_1^2-2c_1c_2^2+4c_3c_1+2c_2^2)e_n^3\\ +(-c_2(c_2^2+2c_1c_3)+5c_1c_4+5c_2c_3-6c_1^2c_4+4c_1^3c_4-c_1^4c_4-6c_1c_2c_3+3c_1^2c_2c_3)e_n^4+O(e_n^5)\}\end{array}}$$

$$=\frac{e_n+2\frac{c_2}{c_1}e_n^2+\frac{(c_2^2+2c_1c_3)}{c_1^2}e_n^3+\frac{(2c_2c_3+2c_1c_4)}{c_1^2}e_n^4}{\left\{\begin{array}{l}1+\frac{(-c_2c_1^2+3c_2c_1)}{c_1^2}e_n+\frac{(c_3c_1^3-3c_3c_1^2-2c_1c_2^2+4c_3c_1+2c_2^2)}{c_1^2}e_n^2\\ +[\frac{(-c_2(c_2^2+2c_1c_3)+5c_1c_4+5c_2c_3-6c_1^2c_4+4c_1^3c_4-c_1^4c_4-6c_1c_2c_3+3c_1^2c_2c_3)}{c_1^2}]e_n^3+O(e_n^4)\end{array}\right\}}$$

$$=\left[e_n+2\frac{c_2}{c_1}e_n^2+\frac{(c_2^2+2c_1c_3)}{c_1^2}e_n^3+\frac{(2c_2c_3+2c_1c_4)}{c_1^2}e_n^4\right]$$
$$\times\left[\begin{array}{l}1-\left\{\frac{(-c_2c_1^2+3c_2c_1)}{c_1^2}e_n+\frac{(c_3c_1^3-3c_3c_1^2-2c_1c_2^2+4c_3c_1+2c_2^2)}{c_1^2}e_n^2\right\}\\ +\frac{(-c_2c_1^2+3c_2c_1)^2}{c_1^4}e_n^2+2\frac{(-c_2c_1^2+3c_2c_1)(c_3c_1^3-3c_3c_1^2-2c_1c_2^2+4c_3c_1+2c_2^2)}{c_1^4}e_n^3\end{array}\right]$$

$$=e_n-\frac{(-c_2c_1^2+3c_2c_1)}{c_1^2}e_n^2-\frac{(c_3c_1^3-3c_3c_1^2-2c_1c_2^2+4c_3c_1+2c_2^2)}{c_1^2}e_n^3$$
$$+\frac{(-c_2c_1^2+3c_2c_1)^2}{c_1^4}e_n^3+2\frac{c_2}{c_1}e_n^2-2\frac{c_2}{c_1}\frac{(-c_2c_1^2+3c_2c_1)}{c_1^2}e_n^3+\frac{(c_2^2+2c_1c_3)}{c_1^2}e_n^3+O(e_n^4)$$
$$=\left[e_n+\left(c_2-\frac{c_2}{c_1}\right)e_n^2+\frac{(3c_3c_1^4-c_1^5c_3-2c_1^3c_2^2-2c_3c_1^3+2c_1^2c_2^2+c_2^2c_1^4)}{c_1^4}e_n^3+O(e_n^4)\right] \qquad (3.7)$$

Therefore,

$$y_n = x_n - \frac{f^2(x_n)}{f(x_n) - f(x_n - f(x_n))}$$

$$= \alpha + e_n - \left[e_n + \left(c_2 - \frac{c_2}{c_1} \right) e_n^2 + \frac{(3c_3c_1^4 - c_1^5c_3 - 2c_1^3c_2^2 - 2c_3c_1^3 + 2c_1^2c_2^2 + c_2^2c_1^4)}{c_1^4} e_n^3 + O(e_n^4) \right]$$

$$y_n = \alpha + \left(\frac{c_2}{c_1} - c_2 \right) e_n^2 - \frac{(3c_3c_1^4 - c_1^5c_3 - 2c_1^3c_2^2 - 2c_3c_1^3 + 2c_1^2c_2^2 + c_2^2c_1^4)}{c_1^4} e_n^3 + O(e_n^4) \quad (3.8)$$

From (3.8), we have

$$f(y_n) = \left(c_2 - c_2c_1 \right) e_n^2 - \frac{(3c_3c_1^4 - c_1^5c_3 - 2c_1^3c_2^2 - 2c_3c_1^3 + 2c_1^2c_2^2 + c_2^2c_1^4)}{c_1^3} e_n^3 + O(e_n^4) \quad (3.9)$$

From (3.2) and (3.9) we have

$$\frac{f(x_n)}{f(x_n) - f(y_n)} = \frac{c_1e_n + c_2e_n^2 + c_3e_n^3 + c_4e_n^4 + c_5e_n^5 + O(e_n^6)}{c_1e_n + c_2c_1e_n^2 + \frac{(3c_3c_1^4 - c_1^5c_3 - 2c_1^3c_2^2 - c_3c_1^3 + 2c_1^2c_2^2 + c_2^2c_1^4)}{c_1^3} e_n^3 + O(e_n^4)}$$

$$= 1 + \left(\frac{c_2}{c_1} - c_2 \right) e_n - \frac{(3c_3c_1^4 - c_1^5c_3 - c_1^3c_2^2 - 2c_3c_1^3 + 2c_1^2c_2^2)}{c_1^4} e_n^2 + O(e_n^3)$$

(3.10)

Combining equations (3.7) and (3.10), we get

$$\frac{f^2(x_n)}{f(x_n) - f(x_n - f(x_n))} \left[\frac{f(x_n)}{f(x_n) - f(y_n)} \right]$$

$$= \left[e_n + \left(c_2 - \frac{c_2}{c_1} \right) e_n^2 + \frac{(3c_3c_1^4 - c_1^5c_3 - 2c_1^3c_2^2 - 2c_3c_1^3 + 2c_1^2c_2^2 + c_2^2c_1^4)}{c_1^4} e_n^3 + O(e_n^4) \right]$$

$$\times \left[1 + \left(\frac{c_2}{c_1} - c_2 \right) e_n - \frac{(3c_3c_1^4 - c_1^5c_3 - c_1^3c_2^2 - 2c_3c_1^3 + 2c_1^2c_2^2)}{c_1^4} e_n^2 + O(e_n^3) \right]$$

(3.11)

with (3.11), (3.1) and (2.5) one can have

$$e_{n+1} + \alpha = e_n + \alpha - \left[e_n + \frac{c_2^2}{c_1} \left(1 - \frac{1}{c_1} \right) e_n^3 \right] \quad (3.12)$$

which yields

$$e_{n+1} \propto e_n^3$$

It shows the method (2.5) has cubic convergence.

IV. NUMERICAL EXAMPLES

We consider few numerical examples considered by Fernando and Weerakoon [13] and by Grewal [1] and the method (2.5) is compared with the methods (1.2), (1.3), (1.6) and (2.1). The computational results are tabulated below and the results are correct up to an error less than 0.5×10^{-20}.

Table 1: *Numerical Comparison*

Function	x_0	number of iteration for each method					ROOT
$f(x)$		**1.2**	**1.3**	**1.6**	**2.1**	**2.5**	
(1) x^3+4x^2-10	1.8	6	6	4	3	3	1.365230013414097
(2) $\cos(x)-x$	4	41	7	8	5	4	0.739085133215161
(3) $(x-1)^3-1$	1.8	6	4	4	3	3	2
(4) $e^{x-1}+x-3$	0	8	5	5	4	4	1.442854401002388
(5) $x\log_{10}x-1.2$	0.5 2.5	7 5	7 5	6 4	4 3	3 2	2.740646095973693
(6) $2x-\log_{10}x-7$	3 4	5 4	4 3	4 3	2 2	2 2	3.789278248444742
(7) $xe^x-\cos x$	0	8	5	6	5	4	0.517757363682458
(8) $e^x\sin x-1$	-0.2	8	5	6	5	5	0.588532743981861
(9) $e^x-1.5-\tan^{-1}x$	-7	7	4	6	4	4	-14.101269772739964

CONCLUSION

With the number of iterations and the root of the respective equation tabulated for each of the methods, we conclude that the method (2.5) which does not require evaluation of the derivatives of the function, has the same or better rate of convergence compared to the methods considered in this paper.

REFERENCES

[1] GREWAL, B.S. *Higher Engineering Mathematics 42nd Edition*, Khanna Publishers, 2012.
[2] MARCHUK, G.I. *Methods of Numerical Mathematics*, springer – Verlag, Newyork, 1982.
[3] ANTI, H.M. *Numerical Methods for Scientists and Engineer*, Tata McGraw – Hill Publishing Company Ltd. New Delhi, 1991.

[4] MATHEWS, J.H. FINK, K. D. *Numerical Methods using Matlab*. Pearson/Prentice Hall New Jersey, 2004.

[5] MATHEWS, J.H. *Numerical Methods for Mathematics*, Science and Engineering, Prentice Hall of India Pvt. Ltd., New Delhi, 2001.

[6] RICE, J.R. *Numerical Methods, Software and Analysis*, McGraw – Hill International Editions, Singapore Computer Science Series, 1987.

[7] JAIN, M.K. , IYENGAR, S.R.K., JAIN, R.K. *Numerical Methods of Scientific and Engineering Computation,* New age international Publisher, New Delhi, India, 2004.

[8] JAIN, M.K. *Numerical Solution of Differential Equations*, 2^{nd} ed. Wiley Easterns Ltd., New Delhi, 1984.

[9] HERNÁNDEZ, M.A. Newton-Raphson's method and convexity Zb. Rad. *Prirod.-Mat. Fak. Ser. Mat*., vol. 22, no. 1, pp. 159-166, 1992.

[10] HAMMING, R.W. *Introduction to Applied Numerical Analysis*, International Student Edition, McGraw Hill Koga Kusha, Ltd, 1971.

[11] SASTRY, S. S. *Introductory Methods of Numerical Analysis*, Prentice – Hall of India, New Delhi, 2005.

[12] RAO, S.S. *Applied Numerical Methods for Engineers and Scientists*, Prentice – Hall, New Jersey, 2002.

[13] T.G.I FERNANDO AND S. WEERAKOON. A Variant of Newton's Method with Accelerated Third-Order Convergence. *Appli. Math. Lett*. 13 87-93.(2000)

[14] KUMAR, V.B., VATTI , KONERU, S.R. *Extrapolated Accelerated Gauss – Seidel Methods*, *Math. Comp,* vol.21, 1987.

[15] KUMAR VATTI, V.B., KUMAR MYLAPALLI, MR.M.S., KUMARI KATRAGADDA, MRS. A. *Extrapolated Newton – Raphson Method*, *Journal of Math Edu*. Vol. XLIII, no.2, June, 2009.

[16] KUMAR VATTI, V. B., SHOURI DOMINIC AND MOUNICA, Extrapolated Newton's Method Free from Second Derivative. *Acta Eng Intl*, vol. no. 1, pp. 39- 43, 2014.